



IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
Re: Appeal to the Board of Patent Appeals and Interferences

AF  
JFW

In re PATENT application of  
DODRILL et al.

Application No. 09/605,848

Filed: June 29, 2000

Title: IP Unified Agent Using An XML Voice Enabled  
Web Based Application Server



Group Art Unit: 2153

Examiner: Strange

Docket: 95-418

Date: August 14, 2007

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

Sir:

- 1 ☐ **NOTICE OF APPEAL:** Applicant hereby appeals to the Board of Patent Appeals and Interferences from the last decision (not Advisory Action) of the Examiner dated February 15, 2007
- 2 ☒ **BRIEF** on appeal in this application attached:
- 3 ☐ An **ORAL HEARING** is respectfully requested under Rule 194 (due two months after Examiner's Answer -- unextendable).
- 4 ☐ Reply Brief is attached in triplicate (due two months after Examiner's Answer -- unextendable).

<b>5. FEE CALCULATION:</b>		Large/Small Entity	
If box 1 above is X'd, see box 12 below <u>first</u> and decide: ..... enter		\$500/250*	\$
If box 2 above is X'd, see box 12 below <u>first</u> and decide: ..... enter		\$500/250*	\$ 500
If box 3 above is X'd, see box 12 below <u>first</u> and decide: ..... enter		\$1000/500*	\$
If box 4 above is X'd, ..... enter nothing		- 0 - (no fee)	
<b>6. Original due date: August 14, 2007</b>			
7. Petition is hereby made to extend the original due date to cover the date this response is filed for which the requisite fee is attached		(1 mo) \$120/\$60 (2 mos) \$450/\$225 (3 mos) \$1020/\$510 (4 mos) \$2160/\$1080	+
8. Enter any previous extension fee paid [ ] previously since above <u>original</u> due date (item 6); [ ] with concurrently filed amendment		-	
9. Subtract line 8 from line 7 and enter: Total Extension Fee			
9a. Terminal Disclaimer Fee			\$
10. TOTAL FEE ATTACHED =			\$ 500.00

11. ☐ \*Fee **NOT** required if/since paid in prior appeal in which the Board of Patent Appeals and Interferences did **not** render a decision on the merits.

**CHARGE STATEMENT:** The Commissioner is hereby authorized to charge any fee specifically authorized hereafter, or any missing or insufficient fee(s) filed, or asserted to be filed, or which should have been filed herewith or concerning any paper filed hereafter, and which may be required under Rules 16-18 (missing or insufficient fee only) now or hereafter relative to this application and the resulting Official document under Rule 20, or credit any overpayment, to our Account/Order No. 50-1130/95-418 for which purpose a duplicate copy of this sheet is attached. This CHARGE STATEMENT does not authorize charge of the issue fee until/unless an issue fee transmittal form is filed.

Atty:   
Edward J. Stemberger  
Reg. No. 36,017  
Tel: (202) 261.1014

Customer No. 23164

Docket No.: 95-418

PATENT

**IN THE UNITED STATES PATENT AND TRADEMARK OFFICE  
BEFORE THE BOARD OF PATENT APPEALS AND INTERFERENCES**

In re Application of

DODRILL et al.

Serial No.: 09/605,848

Filed: June 29, 2000

For: IP UNIFIED AGENT USING AN XML  
VOICE ENABLED WEB BASED APPLICATION SERVER



Group Art Unit: 2153

Examiner: Strange, Aaron

**MAIL STOP: APPEAL BRIEF – PATENTS**

Commissioner for Patents  
P.O. Box 1450  
Alexandria, VA 22313-1450

**APPEAL BRIEF**

Sir:

This is an appeal of the final rejection of claims 1, 3, 5-18, 20, 22-28, 30-39 and 41 in the above-identified patent application.

This Appeal Brief is submitted as required by 37 C.F.R. §41.37.

1. **Real Party in Interest:**

This application is assigned to Cisco Technology, Inc., the real party of interest.

2. **Related Appeals and Interferences:**

There are no other appeals or interferences known to Appellant that will directly affect or be directly affected by or have a bearing on the Board's decision in the pending appeal.

08/15/2007 RHHMED1 00000094 09605848

01 FC:1402

500.00 OP

3. Status of Claims:

Claims 1, 3-20, 22-28, 30-39 and 41 are pending in this application. Claims 4, 19 and 31 would be allowable if rewritten independent format. Claims 1, 3, 5-18, 20, 22-28, 30-39 and 41 stand rejected by the Examiner, and are appealed.

4. Status of any Amendment Filed Subsequent to Final Rejection:

No Amendment was filed in response to the final Office Action of February 15, 2007. A Response to the final Office Action was filed on April 9, 2007.

5. Summary of Claimed Subject Matter:

The claimed subject matter includes independent claims 1, 12, 20, 29 and 39 and dependent claims 3-11, 13-19, 22-27, 30-38 and 41.

Independent claim 1 recites a method in an application server (66 in Fig. 2) for initiating inter-process communication between non-persistent (page 13, line 9) application sessions. The method initiates a first application instance (page 14, lines 27-29, step 300 in Fig. 5) for establishment of an application session between the application server and a first party. It is determined whether a second party is available to receive a message (page 15, lines 21-22, step 314 in Fig. 5) having been established in the application session between the application server and the first party. Based on the determined availability of the second party, a HTML page, originating in the first application instance, is generated having instructions for a browser, in use by the second party, to notify the second party of a new application session for the second party so as to present the message to the second party (page 16, lines 1-9). The generating step includes inserting a uniform resource locator (URL) within the HTML page (page 16, lines 1-3, step 318 in Fig. 5) causing the browser to request interruption of a present application session of the second party, established by another application instance distinct from the first application instance, to create the new application session for the second party (page 15, lines 25-29, step 316 in FIG. 5).

Independent claim 12 recites a method for inter-process communication between non-persistent (page 13, line 9) application instances. The method establishes a first non-persistent application instance serving a first party (page 14, lines 27-29, step 300 in Fig. 5). A second

non-persistent application instance serving a second party is established (page 15, lines 22-23). An HTML page, originating in the first application instance, is generated having instructions for a persistent browser instance in use by the second party, having received the HTML page, to interrupt a present application session having been generated by the second application instance on behalf of the second party and initiate a new application session for the second party (page 15, lines 23-29 and page 16, lines 1-18, steps 316, 318 in Fig. 5).

Independent claim 20 recites an application server (page 9, lines 17-26, 66 in Fig. 3) configured for executing a messaging application. The application server includes an application runtime environment (page 9, line 27 to page 10 line 10, 100 in Fig. 3) configured for dynamically originating and generating in a first application instance between the application server and a first party (page 14, lines 27-29, step 300 in Fig. 5). A hypertext markup language (HTML) document has instructions for a browser to notify a second party of a new application session for the second party (page 16, lines 1-9, step 318 in Fig. 5), based on a determination that the second party using the browser is available to receive the HTML document (page 15, lines 21-24, step 314 in Fig. 5). The application runtime environment is configured to access a common resource (84 in Fig. 3) containing information regarding both the first and second parties (page 15, lines 16-20, Fig. 3). The HTML document has instructions to interrupt a present application session of the second party, established by another application instance distinct from the first application instance, to create the new application session for the second party (page 15, lines 23-29 and page 16, lines 1-18, steps 316, 318 in Fig. 5).

Independent claim 28 recites a computer readable medium (page 14, lines 23-26) having stored thereon sequences of instructions for initiating inter-process communication between non-persistent (page 13, line 9) application sessions, the sequences of instructions including instructions for performing the steps of initiating a first application instance for establishment of an application session between the application server and a first party (page 14, lines 27-29, step 300 in Fig. 5). It is determined whether a second party is available to receive a message having been established in the application session between the application server and the first party (page 15, lines 21-22, step 314 in Fig. 5). Based on the determined availability of the second party (page 15, lines 21-24, step 314 in Fig. 5), a HTML page, originating in the first application instance is generated having instructions for a persistent browser, in use by the second party, to

notify the second party of a new application session for the second party so as to present the message to the second party (page 16, lines 1-9, step 318 in Fig. 5). The generating step includes inserting a uniform resource locator (URL) within the HTML page causing the browser to request interruption of a present application session of the second party, established by another application instance distinct from the first application instance, to create the new application session for the second party (page 15, lines 23-29 and page 16, lines 1-18, steps 316, 318 in Fig. 5).

Independent claim 39 recites an application server (page 9, lines 17-26, 66 in Fig. 3) configured for executing a messaging application. The application server includes means (e.g., application runtime environment, page 9, line 27 to page 10 line 10, 100 in FIG. 3) for dynamically originating and generating in a first application instance between the application server and a first party (page 14, lines 27-29, step 300 in Fig. 5), a hypertext markup language (HTML) document having instructions for a browser to notify a second party of a new application session for the second party so as to present a message from the first party to the second party (page 16, lines 1-9, step 318 in Fig. 5), based on a determination that the second party using the browser is available to receive the message (page 15, lines 21-24, step 314 in Fig. 5). The HTML document has instructions to interrupt a present application session of the second party, established by another application instance distinct from the first application instance, to create the new application session for the second party (page 15, lines 23-29 and page 16, lines 1-18, steps 316, 318 in Fig. 5).

6. Grounds of Rejection to be Reviewed on Appeal:

A. Whether claims 1, 3, 5-7, 9, 12-14, 17, 18, 20, 22-24, 27, 28, 30, 32-34, 36, 39 and 41 are unpatentable under 35 USC §103(a) as obvious over U.S. Patent No. 6,484,196 to Maurille in view of U.S. Patent No. 6,327,622 to Jindal et al.

7. Arguments:

**Claims 1, 3, 5-7, 9, 12-14, 17, 18, 20, 22-24, 27, 28, 30, 32-34, 36, 39 and 41 are patentable under 35 USC §103(a) as not being obvious over U.S. Patent No. 6,484,196 to Maurille in view of U.S. Patent No. 6,327,622 to Jindal et al.**

With regard to the independent claims, claims 1 and 28 recites generating a HTML page, originating in the first application instance, having instructions for a browser, in use by the second party, to notify the second party of a new application session for the second party so as to present a message to the second party. The generating step includes inserting a uniform resource locator (URL) within the HTML page causing the browser to request interruption of a present application session of the second party, established by another application instance distinct from the first application instance, to create the new application session for the second party.

Independent claim 12 recites generating an HTML page, originating in the first application instance, having instructions for a persistent browser instance in use by the second party, having received the HTML page, to interrupt a present application session having been generated by the second application instance on behalf of the second party and initiate a new application session for the second party.

Independent claims 20 and 39 recites that an HTML document has instructions to interrupt a present application session of the second party, established by another application instance distinct from the first application instance, to create the new application session for the second party.

By preserving multiple sessions, parties involved in an instant messaging session, a voice messaging session or a calling session can suspend their present session or activity for a moment to create a new session to exchange audio messages and when finished, can instantly resume the suspended session. Thus, users have freedom of access providing them the means of making and receiving calls and message management anytime and virtually anywhere.

The Examiner has failed to demonstrate that “there was an apparent reason to combine the known elements *in the fashion claimed* by the [claims] at issue [where] this *analysis should be made explicit*.” *KSR Int’l v. Teleflex, Inc.* No. 04-1350, Slip. op. at 14, 82 USPQ2d 1385, 1396 (U.S. Apr. 30, 2007).

The Examiner has the burden of demonstrating that “there was an apparent reason to combine the known elements *in the fashion claimed*.” *KSR Int’l v. Teleflex, Inc.* No. 04-1350, Slip. op. at 14, 82 USPQ2d 1385, 1396. The rejection has failed to establish the analysis as required by the Supreme Court. Rather, the hypothetical combination teaches no more than “the predictable use of prior art elements according to their established functions,” *Id.*, with no

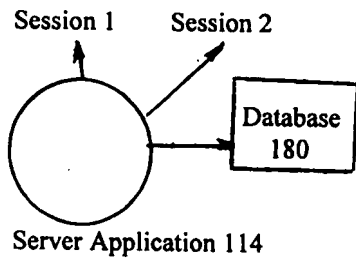
disclosure or suggestion of the claimed features as a whole.

The Examiner admits that Maurille fails to disclose that an application session of the second party is established by another application instance distinct from the first application instance. The Examiner cites Jindal wherein a system is provided for balancing client requests among multiple instances of an application. The Examiner contends that, “This reduces that the load on any single instance of the application, resulting [in] increased performance of the system...” The Examiner further contends that it would have been obvious to one of ordinary skill in the art to modify Maurille “to use multiple instances of the server application to serve sessions from different clients since it would have reduced the load on a single instance of the server application”. Applicants disagree that the combination of Maurille and Jindal suggests the claimed invention.

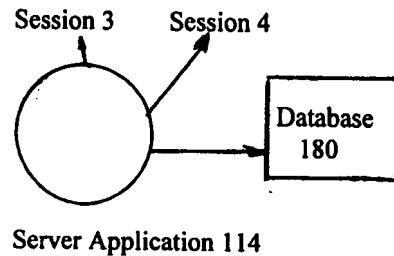
Even if Maurille were modified by employing the teachings of Jindal, the modification would not result in the claimed subject matter, since the combination does not teach or suggest the HTML page originating in the first application instance, requesting interruption of a present application session of the second party, established by another application instance distinct from the first application instance as claimed.

Maurille teaches that a single instance 114 controls all messaging sessions of all users via a database 108 (see Maurille column 6, lines 44-57). Jindal shows in Fig. 2 multiple instances of a program 104a, 104b, 104c for a client 120 for load balancing. With reference to the sketch below, if Maurille was combined with Jindal to include multiple instances of the Maurille’s server application 114, there would merely be another server application 114 and associated database 180. The first and second instances would be distinct and would have no interrupt relation with respect to each other (such as the interruption of the session of the second party as claimed). More particularly, in the combination, session 2 would “interrupt” session 1 or session 4 would “interrupt” session 3 within the **same** application instance. The claims require generating a HTML page originating in the first application instance, requesting interruption of a present application session of the second party, established by another application instance distinct from the first application instance.

### Hypothetical Maurille-Jindal Combination

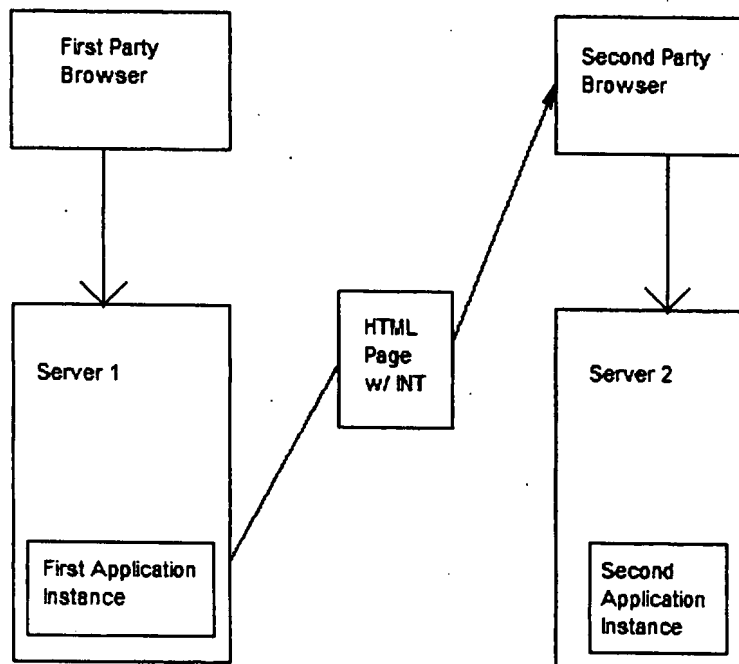


First Instance



Second Instance

In the Advisory Action of April 25 2007, the Examiner presented his own sketch of his proposed hypothetical combination of Maurille and Jindal. The drawing is presented below.



Applicant submits that this sketch IS NOT an accurate representation of the combination of the teachings Maurille and Jindal. The Examiner contends that the HTML "page is generated in the first instance, and sent to the second proxy browser. The page contains instructions that cause the browser to request interruption of the present application session between the second



party and the second application instance.” These features are simply not taught or suggested by Maurille and Jindal. The Examiner is using improper hindsight in creating this sketch. “It is impermissible to use the claimed invention as an instruction manual or ‘template’ to piece together the teachings of the prior art so that the claimed invention is rendered obvious.” *In re Fritch*, 23 USPQ2d 1780, 1784 (Fed. Cir. 1992). “A factfinder should be aware, of course, of the distortion caused by hindsight bias and must be cautious of arguments reliant upon ex post reasoning.” *KSR Int’l v. Teleflex, Inc.* No. 04-1350, Slip. op. at 17, 82 USPQ2d 1385, 1397 (U.S. Apr. 30, 2007) (citations omitted).

Establishing a second application instance as a result of the combination of Maurille and Jindal in no way teaches or suggests a HTML page, established in the first application instance to interrupt a present application session of the second party, established by another application instance distinct from the first application instance as claimed. Again, since Maurille teaches that an instance 114 controls all messaging sessions of all users via a database 108. Jindal shows in Fig. 2 multiple instances of a program 104a, 104b, 104c for a client 120. Thus, and even if there were multiple instances 114 in Maurille, any interruption would occur within the **same** instance and between sessions, as indicated in the Hypothetical Maurille-Jindal Combination sketch above.

Hence, the hypothetical combination teaches no more than “the predictable use of prior art elements according to their established functions,” *KSR Int’l v. Teleflex, Inc.* No. 04-1350, Slip. op. at 14, 82 USPQ2d 1385, 1396, with no disclosure or suggestion of the claimed features as a whole. Hence the claimed interruption of a present application session of the second party, established by another application instance distinct from the first application instance provides “more than the predictable use of prior art elements according to their established functions”. *KSR Int’l v. Teleflex, Inc.*, Slip op. at 13, 82 USPQ 2d 1385, 1396 (U.S. Apr. 30, 2007).

For these and other reasons, the §103 rejection of independent claims 1, 12, 20, 28 and 39 and the claims that depend there-from should be withdrawn.

### Conclusion

For the reasons set forth above, it is clear that Appellant’s claims 1, 3, 5-7, 9, 12-14, 17, 18, 20, 22-24, 27, 28, 30, 32-34, 36, 39 and 41 are patentable over the references applied.

Accordingly the appealed claims 1, 3, 5-18, 20, 22-28, 30-39 and 41 should be deemed patentable over the applied references. It is respectfully requested that this appeal be granted and that the Examiner's rejections be reversed.

To the extent necessary, Appellant petitions for an extension of time under 37 C.F.R. 1.136 and 37 C.F.R. 41.37(e). Please charge any shortage in fees due in connection with the filing of this paper, including any missing or insufficient fees under 37 C.F.R. 1.17(a) or 41.20(b)(2), to Deposit Account No. 50-0687, under Order No. 62-337, and please credit any excess fees to such deposit account.

Respectfully submitted,

A handwritten signature in black ink, appearing to read "E. Stemberger", with a stylized, flowing script.

Edward J. Stemberger  
Registration No. 36,017

Customer No. 23164

**August 14, 2007**

## CLAIM APPENDIX – CLAIMS ON APPEAL

1. (Previously Presented) A method in an application server for initiating inter-process communication between non-persistent application sessions, the method comprising:

initiating a first application instance for establishment of an application session between the application server and a first party;

determining whether a second party is available to receive a message having been established in the application session between the application server and the first party; and

based on the determined availability of the second party, generating a HTML page, originating in the first application instance, having instructions for a browser, in use by the second party, to notify the second party of a new application session for the second party so as to present the message to the second party,

wherein the generating step includes inserting a uniform resource locator (URL) within the HTML page causing the browser to request interruption of a present application session of the second party, established by another application instance distinct from the first application instance, to create the new application session for the second party.

2. Cancelled

3. (Previously Presented) The method of claim 1, further comprising generating a new session identifier that specifies the new application session for the second party, wherein the URL includes the new session identifier for interrupting the present session of the second party with the new application session.

4. (Previously Presented) The method of claim 3, further including initiating a second application instance for execution of the new application session for the second party based on a server-side data record configured for storing a state of the new application session and selected based on the new session identifier, in response to receipt of the URL from the browser.

5. (Previously Presented) The method of claim 1, wherein the HTML page includes a prompt enabling the second party to respond to the message.

6. (Original) The method of claim 1, wherein the determining step includes accessing a registry locally accessible by the application server, and the method further including updating the registry to indicate that the first party is available for messaging operations.

7. (Original) The method of claim 1, further including storing the message in a data store of the second party.

8. (Original) The method of claim 7, wherein storing of the message is performed in accordance with IMAP protocol.

9. (Original) The method of claim 1, further including accessing attribute information of the second party to determine whether the second party authorizes receipt of the message from the first party.

10. (Original) The method of claim 9, wherein the attribute accessing step includes accessing a database server according to LDAP protocol.

11. (Original) The method of claim 1, wherein the message is a voice message and the HTML page includes instructions for playing the voice message.

12. (Previously Presented) A method for inter-process communication between non-persistent application instances, the method comprising:

establishing a first non-persistent application instance serving a first party;

establishing a second non-persistent application instance serving a second party; and

generating an HTML page, originating in the first application instance, having instructions for a persistent browser instance in use by the second party, having received the HTML page, to interrupt a present application session having been generated by the second

application instance on behalf of the second party and initiate a new application session for the second party.

13. (Original) The method of claim 12, further including accessing, by at least one of the first and second application instances, a common resource over an IP network.

14. (Original) The method of claim 13, wherein the common resource is a registry, the method including accessing the registry to determine whether the second party is currently active in the second application instance.

15. (Original) The method of claim 13, wherein the common resource is a data store for storing attribute information of each of the first and second parties, the method including accessing the data store in accordance with LDAP protocol.

16. (Previously Presented) The method of claim 13, wherein the common resource is a messages store for storing messages for each of the first and second parties, the method including accessing the message store in accordance with IMAP protocol.

17. (Original) The method of claim 12, wherein the first application instance is established in first application server and the second application instance is established in a second application server.

18. (Original) The method of claim 13, wherein the common resource is accessible via an application programming interface (API).

19. (Original) The method of claim 12, further including initiating an application instance for execution of the new application session for the second party based on a server-side data record configured for storing a state of the new application session and selected based on the new session identifier, in response to receipt of the HTML page from the browser.

20. (Previously Presented) An application server configured for executing a messaging application, the application server including:

an application runtime environment configured for dynamically originating and generating in a first application instance between the application server and a first party, a hypertext markup language (HTML) document having instructions for a browser to notify a second party of a new application session for the second party, based on a determination that the second party using the browser is available to receive the HTML document, the application runtime environment being configured to access a common resource containing information regarding both the first and second parties,

wherein the HTML document has instructions to interrupt a present application session of the second party, established by another application instance distinct from the first application instance, to create the new application session for the second party.

21. Cancelled

22. (Previously Presented) The application server of claim 20, wherein the HTML document includes a prompt enabling the second party to respond to the message.

23. (Original) The application server of claim 20, wherein the common resource includes a registry and the application runtime environment is configured to access the registry and to update the registry to indicate that the first party is available for messaging operations.

24. (Original) The application server of claim 20, wherein the application runtime environment is configured to access the common resource via an application programming interface (API).

25. (Original) The application server of claim 20, wherein the common resource includes user attribute information stored in a database server in accordance with LDAP protocol and the application runtime environment is configured to access the database server.

26. (Original) The application server of claim 20, wherein the common resource includes a message store for storing the message in accordance with IMAP protocol and the application runtime environment is configured to access the message store.

27. (Original) The application server of claim 20, wherein the common resource includes a registry and the application runtime environment is configured to access the registry and to determine whether the second party is available to receive the message.

28. (Previously Presented) A computer readable medium having stored thereon sequences of instructions for initiating inter-process communication between non-persistent application sessions, the sequences of instructions including instructions for performing the steps of:

initiating a first application instance for establishment of an application session between the application server and a first party;

determining whether a second party is available to receive a message having been established in the application session between the application server and the first party; and

based on the determined availability of the second party, generating a HTML page, originating in the first application instance having instructions for a persistent browser, in use by the second party, to notify the second party of a new application session for the second party so as to present the message to the second party,

wherein the generating step includes inserting a uniform resource locator (URL) within the HTML page causing the browser to request interruption of a present application session of the second party, established by another application instance distinct from the first application instance, to create the new application session for the second party.

29. Cancelled

30. (Previously Presented) The medium of claim 28, further comprising generating a new session identifier that specifies the new application session for the second party, wherein the URL includes the new session identifier for interrupting a present session of the second party

with the new application session.

31. (Previously Presented) The medium of claim 30, further including initiating a second application instance for execution of the new application session for the second party based on a server-side data record configured for storing a state of the new application session and selected based on the new session identifier, in response to receipt of the URL from the browser.

32. (Previously Presented) The medium of claim 28, wherein the HTML page includes a prompt enabling the second party to respond to the message.

33. (Original) The medium of claim 28, wherein the determining step includes accessing a registry locally accessible by the application server, and the method further including updating the registry to indicate that the first party is available for messaging operations.

34. (Original) The medium of claim 28, further including storing the message in a data store of the second party.

35. (Original) The medium of claim 34, wherein storing of the message is performed in accordance with IMAP protocol.

36. (Original) The medium of claim 28, further including accessing attribute information of the second party to determine whether the second party authorizes receipt of the message from the first party.

37. (Original) The medium of claim 36, wherein the attribute accessing step includes accessing a database server according to LDAP protocol.

38. (Original) The medium of claim 28, wherein the message is a voice message and the



HTML page includes instructions for playing the voice message.

39. (Previously Presented) An application server configured for executing a messaging application, the application server including:

means for dynamically originating and generating in a first application instance between the application server and a first party, a hypertext markup language (HTML) document having instructions for a browser to notify a second party of a new application session for the second party so as to present a message from the first party to the second party, based on a determination that the second party using the browser is available to receive the message,

wherein the HTML document has instructions to interrupt a present application session of the second party, established by another application instance distinct from the first application instance, to create the new application session for the second party.

40. Cancelled

41. (Previously Presented) The application server of claim 39, wherein the HTML document includes a prompt enabling the second party to respond to the message.

## EVIDENCE APPENDIX

Not Applicable

## RELATED PROCEEDINGS APPENDIX

Not Applicable